CHAPTER 2

Fundamental Concepts

How do machines learn?

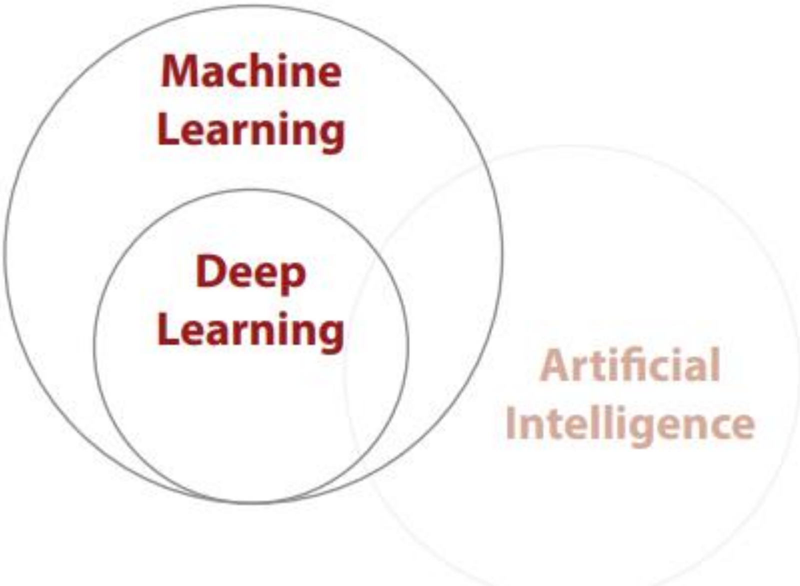
IN THIS CHAPTER

“Machine Learning will cause every successful IPO win in 5 years” - Eric Schmidt (Google Chairman, 2016)

What is Deep Learning?

Deep Learning is a subfield of methods for Machine Learning

Deep Learning is a subset of Machine Learning, which is a fi eld dedicated to the study and development of machines that can learn (sometimes with the goal of eventually attaining General Artificial Intelligence). In industry, Deep Learning is used to solve practical tasks in a variety of fi elds such as Computer Vision (Image), Natural Language Processing (Text), and Automatic Speech Recognition (Audio). In short, Deep Learning is a subset of methods in the Machine Learning toolbox, primarily leveraging Artificial Neural Networks, which are a class of algorithm loosely inspired by the human brain.



Notice in the figure above that not all of Deep Learning is focused around pursuing Generalized Artif cial Intelligence (i.e. sentient machines like in the movies). In fact, many applications of this technology are applied to solve a wide variety of problems in industry. Th is book seeks to focus on learning the fundamentals of Deep Learning behind both cutting edge research and industry, helping to prepare you for either.

Artificial Intelligence Machine Learning Deep Learning Licensed to Asif Qamar

What is Machine Learning? 11 What is Machine Learning?

> "A field of study that gives computers the ability to learn without being explicitly programmed"
>
> - Arthur Samuel

Given that Deep Learning is a subset of Machine Learning, what is Machine Learning? Most generally, it is what its name implies. Machine Learning is a subfield of Computer Science wherein machines learn to perform tasks for which they were not explicitly programmed. In short, machines observe a pattern and attempt to imitate it in some way which can be either direct or indirect imitation.

machine learning ~= monkey see monkey do

I mention direct and indirect imitation as a parallel to the two main types of machine learning, supervised machine learning and unsupervised machine learning. Supervised machine learning is the direct imitation of a pattern between two datasets. It is always attempting to take an input dataset and transform it into an output dataset. This can be an incredibly powerful and useful capability. Consider the following examples: (input datasets in bold and output datasets in italic)

- Using the pixels of an image to detect the presence or absence of a cat
- Using the movies you've liked to predict movies you may like
- Using someone's words to predict whether they are happy or sad.
- Using weather sensor data to predict the probability of rain.
- Using car engine sensors to predict the optimal tuning settings.
- Using news data to predict tomorrow's stock price.
- Using an input number to predict a number double its size.
- Using a raw audio file to predict a transcript of the audio.

These are all supervised machine learning tasks. In all cases the machine learning algorithm is attempting to imitate the pattern between the two datasets in such a way that it can use one dataset to predict the other. For any example above, imagine if you had the power to predict the output dataset given only the input dataset. This would be profound.

“A field of study that gives computers the ability to learn without being explicitly programmed’” { } - Arthur Samuel machine learning ~= monkey see monkey do Licensed to Asif Qamar 12 Chapter 2 I Fundamental Concepts

Supervised Machine Learning

Supervised Learning transforms one dataset into another.

Supervised Learning is a method for transforming one dataset into another. For example, if we had a dataset of "Monday Stock Prices" which recorded the price of every stock on every Monday for the past 10 years, and a second dataset of "Tuesday Stock Prices" recorded over the same time period, a supervised learning algorithm might try to use one to predict the other.

If we successfully trained our supervised machine learning algorithm on 10 years of Mondays and Tuesdays, then we could predict the stock price of any Tuesday in the future given the stock price on the immediately preceeding Monday. I encourage you to stop and consider this for a moment.

Supervised Machine Learning is the bread and butter of applied Artifi cial Intelligence (i.e. "Narrow AI"). It is useful for taking what we do know as input and quickly transforming it into what we want to know. This allows supervised machine learning algorithms to extend human intelligence and capabilities in a seemingly endless number of ways.

Th e majority of work leveraging machine learning results in the training of a supervised classifi er of some kind. Even unsupervised machine learning (which we will learn more about in a second) is typically done to aid in the development of an accurate supervised machine learning algorithm.

For the rest of this book, we will be creating algorithms that can take input data that is observable, recordable, and by extension knowable and transform it into valuable output data that requires logical analysis. Th is is the power of supervised machine learning.

Tuesday stock prices Monday stock prices supervised learning what we want to know what we know supervised learning Licensed to Asif Qamar Unsupervised Machine Learning 13

Unsupervised Machine Learning

Unsupervised Learning groups your data.

Unsupervised learning shares a property in common with supervised learning. It transforms one dataset into another. However, the dataset that it transforms into is not previously known or understood. Unlike supervised learning, there is no "right answer" that we're trying to get the model to duplicate. We just tell an unsupervised algorithm to "fi nd patterns in this data and tell me about them".

For example, clustering a dataset into groups is a type of unsupervised learning. "Clustering" transforms your sequence of datapoints into a sequence of cluster labels. If it learns 10 clusters, it's common for these labels to be the numbers 1-10. Each datapoint will get assigned to a number based on which cluster it is in. Th us, your dataset turns from a bunch of datapoints into a bunch of labels. Why are the labels numbers? Th e algorithm doesn't tell us what the clusters are. How could it know? It just says "Hey scientist!... I found some structure. It looks like there are some groups in your data. Here they are!".

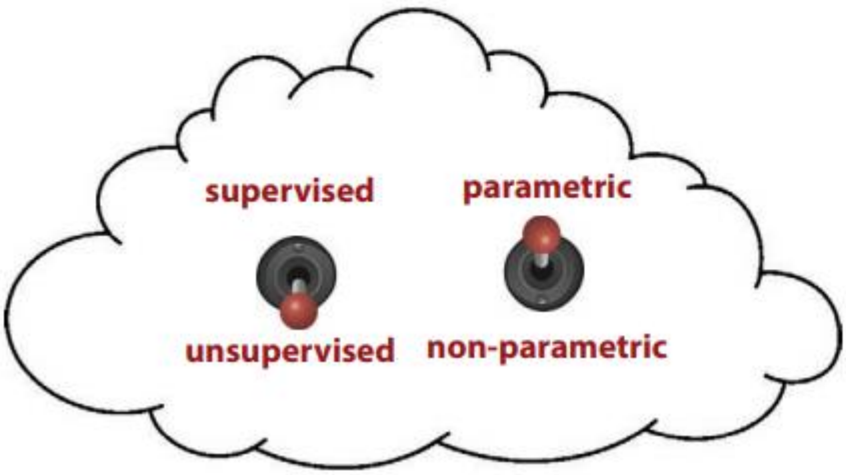
I have good news! Th is idea of clustering is something you can reliably hold onto in your mind as the defi nition of unsupervised learning. Even though there are many forms of unsupervised learning, all forms of unsupervised learning can be viewed as a form of clustering. We will discover more on this later in the book.

Check out the example above. Even though the algorithm didn't tell us what the clusters are named, can you fi gure out how it clustered the words? (1 == cute & 2 == delicious) Later, we will unpack how other forms of unsupervsied learning are also just a form of clustering and why these clusters are useful for supervised learning.

puppies pizza kittens hotdog burger unsupervised learning 1 2 1 2 2 list of cluster labels list of data points unsupervised learning Licensed to Asif Qamar 14 Chapter 2 I Fundamental Concepts

Parametric vs Non-Parametric Learning

Oversimplified: Trial and error learning versus counting and probability Th e last two pages divided all of our machine learning algorithms into two groups, supervised and unsupervised. Now, we're going to discuss another way to divide the same machine learning algorithms into two groups, parametric and non-parametric. So, if we think about our little machine learning cloud, it has two settings:



As you can see, we really have four different types of algorithm to choose from. An algorithm is either unsupervised or supervised and it is either parametric or non parametric. Whereas the previous section on supervision is really about the type of pattern being learned, parametricism is about the way the learning is stored and oft en by extension, the method for learning. First, let's look at the formal definition for parametricism vs non parametricism. For the record, there is still some debate around the exact diff erence.

As an example, let's say the problem was to fi t a square peg into the correct (square) hole. Some humans (such as babies) just jam it into all the holes until it fi ts somewhere (parametric). A teenager, however, might just count the number of sides (4) and then search for the hole with an equal number (non-parametric). Parametric models tend to use "trial and error", where non-parametric models tend to "count". Let's look closer. supervised unsupervised parametric non-parametric

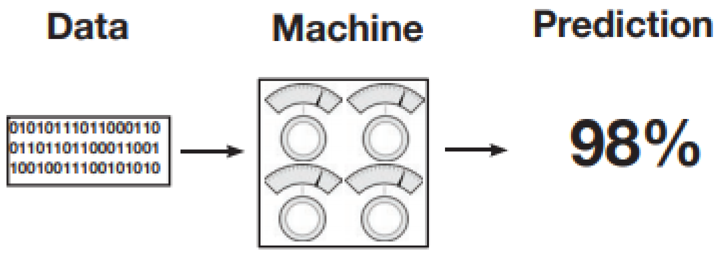
A parametric model is characterized by having a fixed number of parameters whereas a non-parametric model's number of parameters is infinite (determined by data).

Licensed to Asif Qamar Supervised Parametric Learning

Supervised Parametric Learning

Oversimplified: Trial and error learning using knobs

Supervised parametric learning machines are machines with a f xed number of knobs (that's the parametric part), wherein learning occurs by turning the knobs. Input data comes in, is processed based on the angle of the knobs, and is transformed into a prediction.



Learning is accomplished by turning the knobs to diff erent angles. If we're trying to predict the probability that the Red Socks will win the World Series, then this model would fi rst take data (such as sports stats like win/loss record or average number of toes) and make a prediction (such as 98% chance). Next, the model would observe whether or not the Red Socks actually won. Aft er it knew whether they won, our learning algorithm would update the knobs to make a more accurate prediction the next time it sees the same/similar input data.

Perhaps it would "turn up" the "win/loss record" knob if the team's win/loss record was a good predictor. Inversely, it might turn down the "average number of toes" knob if that datapoint wasn't a good predictor. Th is is how parametric models learn!

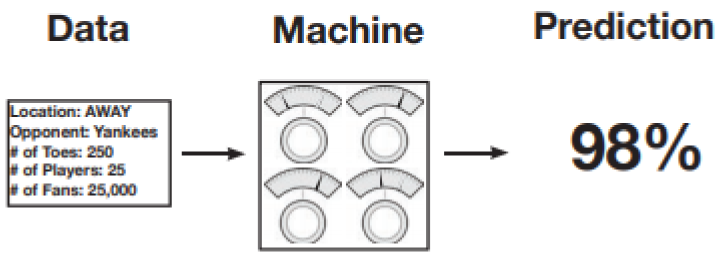
Note that the entirety of what the model has learned can be captured in the positions of the knobs at any given time. One can also think of this type of learning model as a search algorithm. We are "searching" for the appropriate knob confi guration by trying confi gurations, adjusting them, and retrying.

Note further that the notion of trial and error isn't the formal defi nition, but it is a very common (with exceptions) property to parametric models. When there is an arbitrary (but fi xed) number of nobs to turn, then it requires some level of searching to fi nd the optimal confi guration. Th is is in contrast to non-parametric learning, which is oft en "count" based and (more or less) "adds new knobs" when it fi nds something new to count. Let's break down supervised parametric learning into its three steps.

01010111011000110 01101101100011001 10010011100101010 Data Machine Prediction 98% Licensed to Asif Qamar 16 Chapter 2 I Fundamental Concepts

Step 1: Predict

To illustrate supervised parametric learning, let's continue with our sports analogy where we're trying to predict whether or not the Red Socks will win the World Series. Th e fi rst step, as mentioned, is to gather sports statistics, send them through our machine, and make a prediction on the probability that the Red Socks will win.



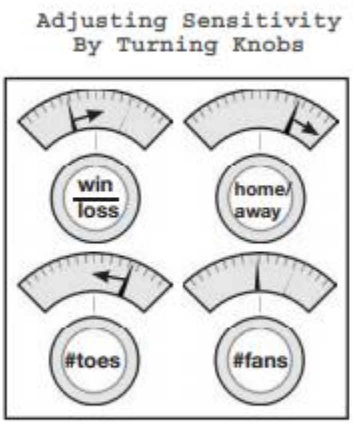
Step 2: Compare to Truth Pattern

The second step is to compare the prediction (98%) with the pattern that we care about (whether the Red Socks won). Sadly, they lost, so our comparison is:

Pred: **98%** **>** Truth: **0%**

Th is step simply recognizes that if our model had predicted 0%, it would have perfectly predicted the upcoming loss of the team. We want our machine to be accurate, which takes us to Step 3.

Step 3: Learn the Pattern win loss home/ away #toes #fans Adjusting Sensitivity By Turning Knobs Location: AWAY Opponent: Yankees # of Toes: 250 # of Players: 25 # of Fans: 25,000 Data Machine Prediction 98% Pred: 98% > Truth: 0% Th is step adjusts the knobs by studying both how much the model missed (98%) and what the input data was (sports stats) at the time of prediction. It then turns the knobs to make a more accurate prediction given the input data. In theory, the next time it saw the same sports stats, the prediction would be lower than 98%. Note that each knob represents the prediction's sensitivity to diff erent types of input data. Th at's what we're changing when we "learn".

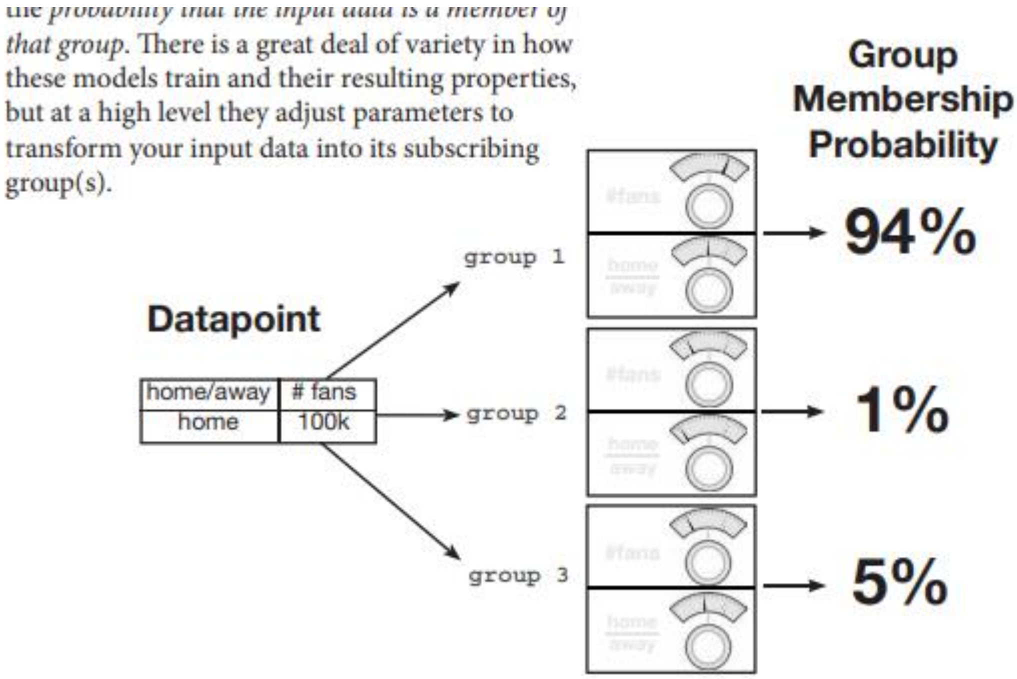


Unsupervised Parametric Learning

Unsupervised parametric learning leverages a very similar approach. Let's walk through the steps at a high level. Remember that unsupervised learning is all about grouping your data. Unsupervised parametric learning uses knobs to group your data. However, in this case, it usually has several knobs for each group, each that maps your input data's affi nity to that particular group (with exception and nuance, this is a high level description). Let's look at home/away home away home home away away away # fans 100k 50k 100k 99k 50k 10k 11k home/away home # fans 100k group 1 group 2 group 3 94% 1% 5% Datapoint Group Membership Probability #fans home away #fans home away #fans home away an example where we assume we want to divide our data into three groups.

home/away	# fans
home	**100k**
away	50k
home	**100k**
home	**99k**
away	50k
away	*10k*
away	*11k*

In the dataset on the right, I have identifi ed three clusters in the data that we might want our parametric model to fi nd. I identifi ed them via formatting as group 1, group 2, and group 3. Let's propagate our fi rst datapoint through a trained unsupervised model below. Notice that it maps most strongly to group one. Each group's machine attempts to transform the input data to a number between 0 and 1, telling us the probability that the input data is a member of that group. Th ere is a great deal of variety in how these models train and their resulting properties, but at a high level they adjust parameters to transform your input data into its subscribing group(s).



Licensed to Asif Qamar 18 Chapter 2 I Fundamental Concepts

Non-Parametric Learning

Oversimplified: Counting based methods

Non-Parametric learning is a class of algorithm wherein the number of parameters is based on data (instead of pre-defined). This lends itself to methods that generally count in one way or another, thus increasing the number of parameters based on the number of items being counted within the data. In the supervised setting, for example, a non-parametric model might count the number of times a particular color of streetlight causes cars to "go". After counting only a few examples, this model would then be able to predict that middle lights always (100%) cause cars to "go" and right lights only sometimes (50%) cause cars to "go".

Notice that this model would have 3 parameters, 3 counts indicating the number of times each colored light turned on and cars "go" (perhaps divided by the number of total observations). If there had been 5 lights, there would have been 5 counts (5 parameters). What makes this simple model non-parametric is this trait wherein the number of parameters changes based on the data (in this case, the number of lights). This is in contrast to parametric models, which start with a set number of parameters and, more importantly, can have more or less parameters purely at the disgretion of the scientist training the model (irrespective of data)

. A close eye might question this idea. Our parametric model from before seemed to have a knob for each input datapoint. In fact, most parametric models still have to have some sort of input that is based on the number of classes in the data. Thus you can see that there is a bit of grayness between parametric and non-parametric algorithms. Even parametric algorithms still are somewhat influenced by the number of classes in the data, even if they are not explicitly counting patterns.

Perhaps this also illumates that parameters is actually a very generic term, referring only to the set of numbers used to model a pattern (without any limitation on how those numbers are used). Counts are parameters. Weights are parameters. Normalized variants of counts or weights are parameters. Correlation coefficients can be parameters. It's simply referring to the set of numbers used to model a pattern. As it happens, Deep Learning is a class of parametric models. We won't be discussing non-parametric models further in this book, but they are a very interesting and powerful class of algorithm. STOP GO GO GO STOP STOP Licensed to Asif Qamar Conclusion

Conclusion In this chapter, we have gone a level deeper into the various flavors of Machine Learning. We have learned that a Machine Learning algorithm is either Supervised or Unsupervised and either Parametric or Non-Parametric. Furthermore, we have explored exactly what makes these 4 different groups of algorithms distinct. We have learned that Supervised Machine Learning is a class of algorithm where we learn to predict one dataset given another and that Unsupervised Learning generally groups a single dataset into various kinds of clusters. We learned that Parametric algorithms have a fixed number of parameters and that Non-Parametric algorithms adjust the number of parameters they have based on the dataset.

Deep Learning leverages Neural Networks to perform both supervised and unsupervised prediction. Up until now, we have stayed at mostly a conceptual level, gaining our bearings on the field as a whole and our place in it. In the next chapter, we will be building our first neural network, and all Subsequent chapters will be project based. So, pull out your Jupyter notebook and let's jump right in!